Tarea Evaluación Derivados Tiempo Discreto

Autor
Manuel de la Llave*

Profesor
Hipòlit Torró

16 de Abril de 2020

*llave@alumni.uv.es

Índice general

1 Portfolio insurance

1.1 Introducción

El objetivo de este ejercicio es diseñar un fondo que replique a un índice y cubrirlo ante posibles bajadas empleando contratos de futuros (técnica conocida como *portfolio insurance*), en otras palabras, vamos a diseñar una *put* sintética; además, valoraremos la *put* que deberíamos haber comprado y evaluaremos así la precisión de nuestra estrategia (*tracking error*). La razón de utilizar una *put* sintética radica en la intención de eliminar parte de los costes de transacción y no generar problemas de liquidez. Por ello, vamos a constituir un fondo que replique el DAX30 (bolsa alemana), usando obviamente futuros sobre el propio DAX30. Hemos elegido el 2 de enero de 2020 como la fecha de inicio de la cobertura y, por simplicidad, hemos supuesto que el capital del fondo es 1000 veces el valor del DAX30 al cierre de ese día (13385930€), único parámetro tomado de manera arbitraria; la cobertura continuará hasta el 9 de abril.

Los cálculos se han realizado usando *R* y se puede consultar el código en el anexo correspondiente. Además, he subido todo el trabajo (datos y script de *R*) a un repositorio GitHub, para que pueda ser fácilmente replicado.

1.2 Datos y nomenclatura

Todos los datos se han obtenido íntegramente de la plataforma Thomson Reuters Eikon (TR en adelante).

- Índice (DAX30):
  - Subyacente (S): precio al cierre del DAX30 desde 02/01/2020 hasta el 09/04/2020
  - Dividendos (q): 3.79859%. Te lo da TR en la página sobre el DAX30. Recogido a día 11/04/2020
  - Volatilidad (σ): hemos empleado los datos históricos del VIX del DAX30 desde 14/04/2000 hasta 31/12/2019, cogiendo la media nos da una volatilidad de $\sigma = 23.0479\%$.

- Futuros sobre el DAX30:
  - Precio de los futuros (F): precio al cierre de los contratos de futuros que vencen en junio de 2020. Como fecha final (T_f) hemos seleccionado el último día de

trading (19/06/2020), suponiendo que hay suficiente liquidez en el mercado.

  - Multiplicador del futuro (K_2): 25

- Otros:

  - Tipo de interés (r): consideramos que el tipo de interés de referencia es el del bono alemán a 10 años. Por un lado, se trata del tipo libre de riesgo de referencia por excelencia en el contexto del mercado europeo, más aún si estamos empleando un índice alemán. Por otro lado, el bono a 10 años es el empleado comúnmente en la literatura académica como tipo libre de riesgo.

    Para hallarlo, hemos cogido el tipo anual medio entre el 31/12/2018 y el 31/12/2019, justo antes de empezar nuestra serie y tan solo un año para que sea lo más relevante posible en el contexto actual. De esta forma, tenemos que $r = -0.214\%$

  - Strike (X): por simplicidad, hemos considerado que el strike de la *put* se corresponde con el primer valor de nuestra serie, $X = 13385.93$€.

  - K_1: número de veces que el fondo contiene al índice, que es 1000.

  - $N(x)$: normal $(0,1)$ del valor x.

  - T: fecha de vencimiento de la opción (09/04/2020). ΔT indica el número de días que quedan hasta el vencimiento. Para anualizarlos, consideramos siempre nºdías/250, también para T_f. Usamos 250 en lugar de 365 porque es la cantidad de días que se negocia en el mercado y es la medida típicamente utilizada.

  - p: opción *put* con *strike* X que vence en T.

1.3 Estrategia

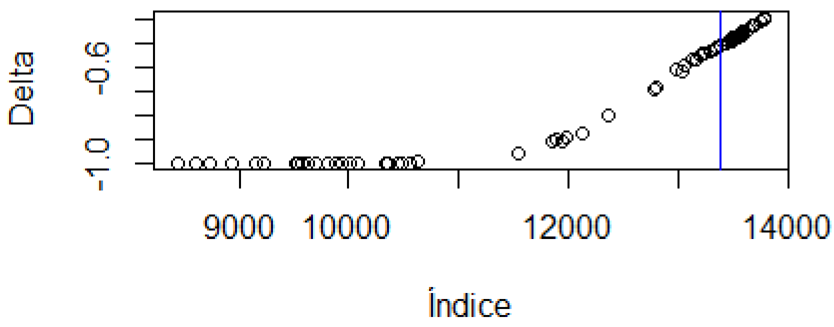
Como hemos comentando, vamos a vender futuros para cubrir el riego de bajada. Necesitamos saber primero cuántos contratos vender, así que obtenemos la cobertura delta antes que nada. Como estamos creando una *put* sintética, lo que en el fondo queremos es $\frac{\partial p}{\partial S} = \delta$, es decir, cómo varía el precio de la *put* ante cambios en el subyacente. De la función de valoración de la *put* tenemos que $\delta = N(d1) - 1$, donde

$$d1 = \frac{ln(S/X) + (r - q + \sigma^2/2)\Delta T}{\sigma\sqrt{\Delta T}} \tag{1}$$

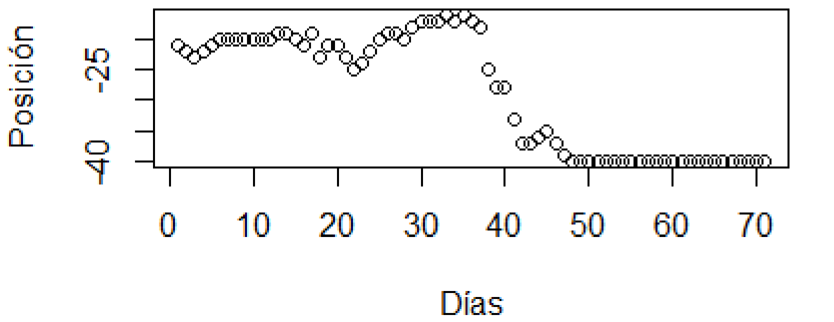
Una vez hallada δ, podemos calcular la posición a tomar en el mercado de futuros como

$$Y = e^{-(r-q)\Delta T_f} \cdot e^{-q\Delta T} \cdot \delta \cdot \frac{K_1}{K_2} \tag{2}$$

Ahora, conforme pase el tiempo y el índice (S) vaya moviéndose, la posición Y irá cambiando. En nuestro caso, la δ resulta de la siguiente manera:



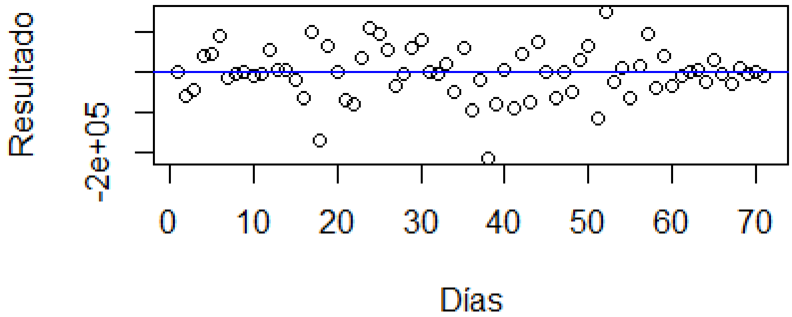
que se corresponde con lo que teóricamente debería suceder.[1] Mientras que la posición:



donde “Posición” hace referencia al número de contratos que debemos comprar (vender en este caso, pues es negativo) en el mercado de futuros, y ´´Días” corresponde al número de días transcurridos desde que comenzamos la cobertura hasta que la finalizamos. Como podemos comprobar, a partir del día 40 (más o menos) el índice cae muy por debajo del *strike* y tanto δ como la posición Y se hacen máximas. Hemos calculado el resultado (beneficios) que obtendrá el fondo día a día teniendo en cuenta los beneficios resultantes

[1] La línea azul marca el strike.

de la revalorización del fondo así como de la liquidación diaria de pérdidas y ganancias:



donde la línea azul representa beneficios 0. Muchos puntos toman valores muy alejados de 0, pero al final del periodo parecen converger y el último día arroja una pérdida de -11150€, que representa un -0.0833% menos del valor inicial del fondo; teniendo en cuenta la gran volatilidad del periodo[2] así como la gran caída que ha experimentado el índice, parece que ha sido una buena cobertura.

1.3.1 *Tracking error*

Si hubiéramos comprado la *put* en lugar de fabricarla sintéticamente, tal vez podríamos haber mejorado nuestra cobertura. Así pues, hemos calculado el valor teórico de una *put* con nuestros parámetros, siguiendo la fórmula de Black-Scholes:

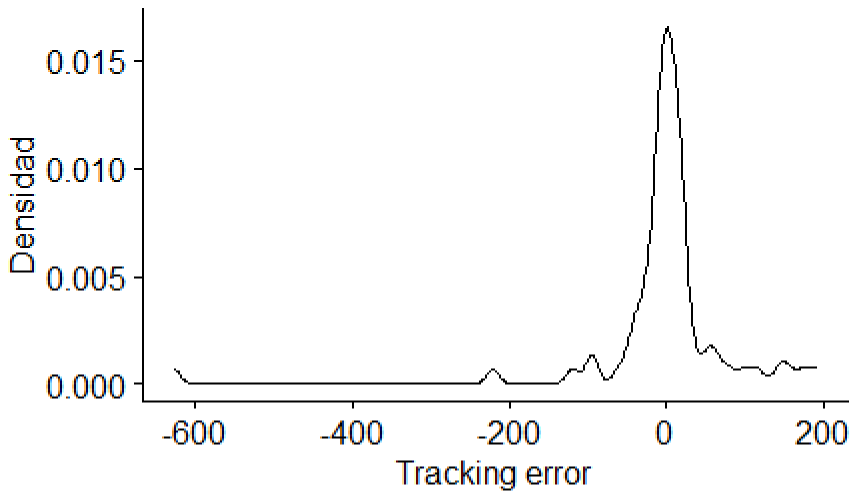
$$p = K_1(Xe^{-rT}N(-d2) - Se^{-qT}N(-d1)) \tag{3}$$

donde multiplicamos por K_1 para adecuar la *put* al tamaño de nuestro fondo. Esto lo hemos comparado con nuestros resultados anteriores para hallar el *tracking error*, de manera que

$$\textit{Tracking error} = \frac{p(t) - p(t+1) + ldpg(t+1)}{ldpg(t+1)}$$

es decir, que lo que ganas/pierdes con la put se vea compensando con lo que ganas/pierdes en la liquidación diaria de pérdidas y ganancias. Gráficamente:

[2] Hay que tener en cuenta que la volatilidad que nosotros consideramos fue la histórica, mientras que en el transcurso de la cobertura, impactó de lleno la crisis del COVID-19, haciendo que la volatilidad fuera mayor. Sin embargo, hemos mantenido los supuestos de esta manera para tratar de replicar fidedignamente lo que hubiéramos hecho partiendo el 2 de enero, y no con el conocimiento *a posteriori*.



A pesar de que aparecen algunos valores extremos (está en porcentajes), la mayoría de días conseguimos un error centrado en el 0, lo que implica una buena cobertura (asumiendo que la put está bien valorada).

2 Valoración y cobertura de una opción asiática de media aritmética

Vamos a valorar una opción asiática mediante simulación y además hallaremos su cobertura delta. El código empleado (usando R), puede consultarse en el anexo correspondiente así como en el repositorio GitHub mencionado a principio del trabajo.

Lo que vamos a hacer es simular m veces la evolución de un activo subyacente S a lo largo de un periodo de tiempo T. El activo en cuestión, puesto que consideramos la probabilidad neutral al riesgo, lo valoraremos de la siguiente forma:

$$S_t = S_{t-1} exp\left\{\left(r - \frac{1}{2}\sigma^2\right)\Delta t + \sigma\sqrt{\Delta t} z_t\right\} \tag{4}$$

donde r es el tipo de interés libre de riesgo, σ es la volatilidad del activo, Δt es la distancia entre los pasos en los que se divide el intervalo y z_t es un proceso browniano. Así pues, tendremos m vectores de subyacente S. Por otro lado, una *call* asiática de media aritmética, tiene el *payoff* $(\overline{S} - K)^+$, con precio de *strike* K; es decir, que usaremos la media de cada vector S que hemos simulado para calcular el precio de una opción, y con los m precios obtenidos, hallaremos la media (precio final) de la opción. En términos matemáticos:

$$c = \frac{e^{-rT}}{m}\sum_{i=1}^{m}\left(\frac{1}{n}\sum_{j=1}^{n} S_{ij} - K\right) \tag{5}$$

Para calcular la delta ($\hat{\Delta}$), realizaremos la misma operación (de hecho, lo haremos todo a la vez como se puede ver en el código), sabiendo que

$$\hat{\Delta} = \begin{cases} \frac{e^{-rT}}{X_0} \frac{1}{n} \sum_{j=1}^{n} X_j & \text{si } \left(\frac{1}{n} \sum_{j=1}^{n} X_j > K \right) \\ 0 & \text{si } \left(\frac{1}{n} \sum_{j=1}^{n} X_j > K \right) \end{cases}$$

Para la simulación hemos empleado como ventana temporal 1 año ($T = 1$), pero lo hemos dividido en 250 pasos para simular una valoración diaria, de la misma manera que hicimos en la tarea 1. Así pues, $\Delta t = 1/250 \approx 0.004$. Como tipo de interés libre de riesgo y volatilidad, vamos a mimetizar el ejercicio anterior y escogeremos $r = -0.00214$, $\sigma = 0.230479$, asímismo el subyacente será el DAX30 y el valor inicial será 13385.93€, mientras que el *strike* será de 13500€, ya que no resulta descabellado pensar a dos de enero de 2020 que el DAX30 esté sobre los 13500 puntos en el siguiente año. Con estos datos y realizando 10000 simulaciones,[3] nos da un valor para la *call* de 1432.48€ y una $\hat{\Delta} = 0.5823$. Dada la aleatoriedad del proceso browniano que hemos generado, no es posible replicar *exactamente* estos valores, pero deberían parecerse.

[3] Advertencia: al replicar el ejercicio, dependiendo del ordenador, 10000 podrían ser muchas iteraciones.

A Código Tarea 1

```
#Paquetes necesarios:
library(dplyr)
library(readxl)
library(dplyr)
library(ggpubr)

#Importamos la serie del DAX30 y de futuros sobre el DAX30 para junio 2020
#sobre la que vamos a trabajar
DAX30_FUT <- read_excel("YOUR PATH/DAX30 + FUT.xlsx")
#Importamos también la serie histórica del VIX para sacar la volatilidad
VIX_DAX30 <- read_excel("YOUR PATH VIX DAX30.xlsx")
#Invertimos el orden de los datos (ordenados por fecha).
datos <- arrange(DAX30_FUT, date)



##PARÁMETROS##

r <- -0.00214 #Tipo de interés medio del bono alemán a 10 años (31DIC18-31DIC19)
q <- 0.0379859 #Dividendos del DAX30 (proporcionado por Thomson Reuters
#a día 11 de abril de 2020)
sd <- mean(VIX_DAX30$vix)/100 #Desviación típica sacada del VIX histórico.
X <- first(datos$dax) #Strike = primer valor del DAX30
S <- datos$dax #Precio cierre DAX30 (subyacente)
f <-  datos$futdax #Precio de los futuros sobre el DAX30 para JUN20
n <- 1000*S #Nominal del fondo
k1 <-  n/S #Veces que la cartera contiene al índice
k2 <- 25 #Multiplicador del futuro
datos$t_f <- as.Date(rep("2020-06-19", times = length(datos$date)))
- as.Date(datos$date) #Días hasta vencimiento del futuro
t_f <- as.numeric(datos$t_f/250) #Anualizamos los días y los convertimos en números
datos$t <- as.Date(rep("2020-04-09", times = length(datos$date)))
- as.Date(datos$date) #Días hasta vencimiento de la opción
t <- as.numeric(datos$t/250) #Anualizamos los días y los convertimos en números
```

```
##FUNCIONES##
d1 <- (log(S/X) + (r - q + sd^2/2)*t)/(sd*sqrt(t))
delta <- pnorm(d1) - 1 #Cobertura delta
posc <-  round(exp(-(r-q)*t_f) *exp(-q*t)*delta*(k1/k2))  #Posición en futuros
ldpg <- 0:(length(datos$futdax)-1) #Creamos un vector ldpg
for (i in 2:71) {
  ldpg[1] <- 0
  ldpg[i] <- (f[i]-f[i-1])*k2*(posc[i])
} #Liquidación diaria de pérdidas y ganancias
perd <- 0:(length(datos$futdax)-1) #Creamos un vector de pérdidas
for (i in 2:71) {
  perd[1] <- 0
  perd[i] <- n[i]-n[i-1]
}#Pérdida del fondo día a día.
rdo <- perd+ldpg #Resultado día a día de las ganancias y pérdidas.


###Para calcular el valor de la put (p)
d2 <- d1 - sd*sqrt(t)
nd1 <- pnorm(-d1)
nd2 <- pnorm(-d2)
p <- 1000*(X*exp(-r*t)*nd2 - S*exp(-q*t)*nd1) #Valoración de la put.
#Multiplicamos por 1000 porque el nominal es 1000 veces X
track_err <- 0:(length(datos$futdax)-1) #Creamos un vector track_err
for (i in 2:length(datos$futdax)) {
  track_err[1] <- 0
  track_err[i] <- ((p[i-1]-p[i]+ldpg[i])*100) /(ldpg[i])
} #Tracking error en porcentaje.


##OTROS##
#Gráfico de delta sobre el subyacente.
plot(S, delta, xlab = "Índice", ylab = "Delta")
abline(v=X, col="blue")
```

```
#Gráfico para observar la posición en el mercado de futuros.
plot(posc, xlab = "Días", ylab = "Posición")

#Gráfico para observar el resultado de la cobertura.
plot(rdo, xlab = "Días", ylab = "Resultado")
abline(h=0, col = "blue")

#Gráfico del tracking error.
ggdensity(track_err, xlab = "Tracking error", ylab = "Densidad")
```

B Código Tarea 2

```
#Parámetros:
t <- 1 #Intervalo temporal, 1 año.
n <- 250 #Pasos en los que se divide el intervalo (lo pasamos a días)
m <- 10000 #Cantidad de simulaciones realizadas.
dt <- t/n #Aproximación del diferencial de t.
S_1 <- 13385.93 #Valor inicial del subyacente.
K <- 13500 #Strike de la opción.
r <- -0.00214$ #Tipo de interés.
sd <- 0.230479 #Volatilidad.

#Dummies para almacenar información en el bucle:
ddelta <- 0 #Dummy delta.
dcall<- 0 #Dummy call.
payoff <- rep(0, m) #Vector que contendrá los pagos de la opción.
z <- rep(0, n) #Vector que contendrá al browniano.
S <- rep(S_1, n) #Vector que contendrá el subyacente.

#Primero creamos el bucle perteneciente a las simulaciones
for (j in 1:m) {
  #Dentro, metemos otro bucle que nos va a dar el vector
  #S, desde S[1] hasta S[n=250].
  for (i in 2:n){
```

```
    z[i] <- rnorm(1, mean = 0, sd = 1) #Browniano.
    S[i] <- S[i-1]*exp((r-(sd^2)/2)*dt +sd*sqrt(dt)*z[i])#Black-Scholes
  }
  #Una vez acabado el bucle, hallamos la función de pagos
  #mediante el siguiente condicional, que nos dará el pago
  #para cada vector S, m veces. Cada S es diferente gracias
  #a la aleatoriedad del browniano, por lo que cada pago
  #también será diferente. Tendremos 100 pagos diferentes en total.
  if (mean(S) > K) {payoff[j] <- mean(S) - K;
                    ddelta <- ddelta + mean(S)} #Acumulamos el valor medio de S.
  else {payoff[j] <- 0} #El condicional hace el papel de función indicatriz.
  dcall <- dcall + payoff[j] #Acumulamos el pago.
}

delta <- (ddelta/(m*S_1))*exp(-r*t) #Valor de delta.
call <- (dcall/m)*exp(-r*t) #Precio de la call.
```